Defense Technical Information Center Compilation Part Notice

ADP011995

TITLE: Interpolating Involute Curves

DISTRIBUTION: Approved for public release, distribution unlimited

This paper is part of the following report:

TITLE: International Conference on Curves and Surfaces [4th], Saint-Malo, France, 1-7 July 1999. Proceedings, Volume 2. Curve and Surface Fitting

To order the complete compilation report, use: ADA399401

The component part is provided here to allow users access to individually authored sections of proceedings, annals, symposia, etc. However, the component should be considered within the context of the overall compilation report and not as a stand-alone technical report.

The following component part numbers comprise the compilation report:
ADP011967 thru ADP012009

Interpolating Involute Curves

Mitsuru Kuroda and Shinji Mukai

Abstract. We propose a straightforward method for designing an interpolating involute curve whose radius of curvature is piecewise linear or quadratic with respect to winding angle. Designers can specify and control the curvature radius profile to a certain extent. End radii of a circle involute are solved in terms of end tangent angles, and a G^1 involute curve is derived by the Hermite interpolation. For G^2 and G^3 involute curves, relevant nonlinear equations are solved by the Newton-Raphson method. NC machines with an involute generator can draw the resulting curves with "reduced data".

§1. Introduction

We present a new method for designing two kinds of smooth interpolating curves, smooth in the sense of consisting of less segments with continuous monotone curvature radius plot. In the method, we derive a G^2 (curvature continuous) involute of circular arcs or a G^3 involute of circle involute arcs through describing its radius of curvature that is piecewise linear or quadratic with respect to winding angle.

"The most important curve in engineering is arguably the circle involute ... it has played key historical roles in a variety of scientific and technological applications" [3]. This curve has excellent shape properties which make it interesting for CAGD (Computer Aided Geometric Design). One can draw the involute curve manually with simple equipment if necessary. NC (Numerical Control) machines with an involute generator are available [1,5].

In our straightforward design method, designers can specify tangents and curvatures at junction points, and control the curvature profile directly to a certain extent. End radii of a circle involute are solved in terms of the end tangent angles, and so an interpolating G^1 involute curve is derived span by span by the two-point Hermite interpolation. For G^2 and G^3 involute curves, continuity conditions and other requirements lead to a system of nonlinear equations. We solve this equation system by the Newton-Raphson method, using initial values from the conventional C^2 cubic spline curve. We obtain examples of the curves satisfying additional requirements, and illustrate the properties of the newly developed curves.

Curve and Surface Fitting: Saint-Malo 1999

Albert Cohen, Christophe Rabut, and Larry L. Schumaker (eds.), pp. 273–280.

ISBN 0-8265-1357-3.

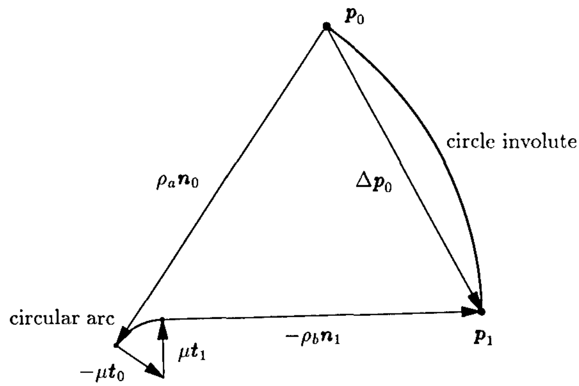


Fig. 1. Circle involute and its evolute.

§2. Circle Involute Arc

A planar curve $\boldsymbol{r}(s)$ is expressed as

$$\boldsymbol{r}(s) = \boldsymbol{p}_0 + \int_0^s \begin{pmatrix} \cos\theta \\ \sin\theta \end{pmatrix} ds, \quad -\infty < \theta < \infty, \tag{1}$$

where s is arclength from the starting point $\boldsymbol{p}_0$ and θ is a winding angle (the angle between tangent vector and the direction of the x axis). The following relations hold among the curve $\boldsymbol{r}$, unit tangent vector $\boldsymbol{t}$, unit normal vector $\boldsymbol{n}$ and radius of curvature ρ:

$$\frac{d^2\boldsymbol{r}}{ds^2} = \frac{d\boldsymbol{t}}{ds} = \frac{d\theta}{ds}\boldsymbol{n} = \frac{1}{\rho}\boldsymbol{n}. \tag{2}$$

The radius of curvature of circle involute is proportional with respect to θ:

$$\rho = \frac{ds}{d\theta} = \rho_a + \mu(\theta - \theta_0), \tag{3}$$

$$\mu \equiv \frac{\rho_b - \rho_a}{\Delta\theta_0} = const.,$$

$$\rho(\theta_0) = \rho_a, \quad \rho(\theta_1) = \rho_b,$$

where μ is the radius of circular arc that is the evolute of $\boldsymbol{r}$, and Δ is the forward difference operator defined by $\Delta z_i \equiv z_{i+1} - z_i$. We change the variable of the expression (1) from s to θ by the relation (3) and integrate it:

$$\Delta \boldsymbol{p}_0 = \int_{\theta_0}^{\theta_1} \rho \boldsymbol{t} d\theta = \rho_a \boldsymbol{n}_0 + \mu(\boldsymbol{t}_1 - \boldsymbol{t}_0) - \rho_b \boldsymbol{n}_1, \tag{4}$$

$$\boldsymbol{r}(\theta_0) = \boldsymbol{p}_0, \quad \boldsymbol{t}(\theta_0) = \boldsymbol{t}_0, \quad \boldsymbol{n}(\theta_0) = \boldsymbol{n}_0,$$
$$\boldsymbol{r}(\theta_1) = \boldsymbol{p}_1, \quad \boldsymbol{t}(\theta_1) = \boldsymbol{t}_1, \quad \boldsymbol{n}(\theta_1) = \boldsymbol{n}_1.$$

The vector equation (4) is understood easily as in Fig. 1.

The expression (4) is a system of equations with respect to unknowns ρ_a and ρ_b. We can solve this as follows:

$$\Delta \boldsymbol{p}_0 \equiv L_0 \begin{pmatrix} \cos\phi_0 \\ \sin\phi_0 \end{pmatrix},$$

$$\begin{aligned} \rho_a &= \frac{\Delta\theta_0 \cos(\theta_1 - \phi_0) + \sin(\theta_0 - \phi_0) - \sin(\theta_1 - \phi_0)}{-2 + 2\cos\Delta\theta_0 + \Delta\theta_0 \sin\Delta\theta_0} L_0, \\ \rho_b &= \frac{\Delta\theta_0 \cos(\theta_0 - \phi_0) + \sin(\theta_0 - \phi_0) - \sin(\theta_1 - \phi_0)}{-2 + 2\cos\Delta\theta_0 + \Delta\theta_0 \sin\Delta\theta_0} L_0. \end{aligned} \tag{5}$$

Since the involute arc is given in terms of start and end points as well as corresponding tangent vectors by the expression (5), we can obtain a G^1 involute curve by the Hermite interpolation which satisfies the equations

$$\boldsymbol{r}(\theta_i) = \boldsymbol{p}_i, \quad i = 0, 1, \ldots, n. \tag{6}$$

Radii of segments of the evolute of $\boldsymbol{r}$ are rewritten as

$$\mu_i = \frac{\cos(\theta_i - \phi_i) - \cos(\theta_{i+1} - \phi_i)}{-2 + 2\cos\Delta\theta_i + \Delta\theta_i \sin\Delta\theta_i} L_i, \quad i = 0,\ 1, \ldots, n-1.$$

§3. Interpolating G^2 Involute Curve

We can also derive an interpolating G^2 involute curve of circular arcs. Using the equation (5), we can solve the following nonlinear equation system (7) with respect to unknowns $\theta_0,\ \theta_1, \ldots, \theta_n$ by the Newton-Raphson method:

$$\rho_i \equiv \rho(-\theta_i) = \rho(+\theta_i), \quad i = 1,\ 2,\ \ldots, n-1. \tag{7}$$

However, the Jacobian matrix necssary for the method makes a programming code long and convergence relatively slow. Therefore, from the practical point of view, we prefer to solve the following equations (8) based on the equation (4) directly. Adding unknowns $\rho_0,\ \rho_1, \ldots, \rho_n$ to the previous ones $\theta_0,\ \theta_1, \ldots, \theta_n$, we get

$$\rho_i \boldsymbol{n}_i + \mu_i \Delta \boldsymbol{t}_i - \rho_{i+1} \boldsymbol{n}_{i+1} = L_i \begin{pmatrix} \cos\phi_i \\ \sin\phi_i \end{pmatrix}, \quad i = 0,\ 1, \ldots, n-1, \tag{8}$$

$$\mu_i = \frac{\Delta\rho_i}{\Delta\theta_i}, \quad i = 0,\ 1, \ldots, n-1.$$

Fig. 2 shows an example of interpolating G^2 involute curves and its profile of curvature and radius of curvature. Initial values were from the conventional C^2 cubic splines. The Newton-Raphson method converged after three iterations. In spite of the unpleasant configuration of data points, the curve derived is quite smooth. Its evolute curve (circular arcs) is G^1 continuous except for two cusp points that correspond to extremal points of the radius of curvature.

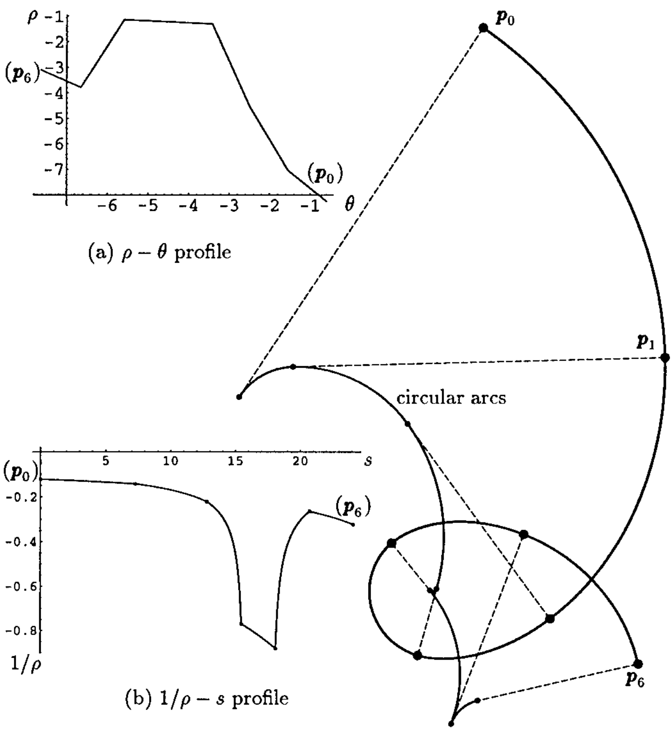


Fig. 2. Interpolating G^2 involute curve.

§4. Interpolating G^3 Involute Curve

Expressing ρ as a quadratic B-spline function of variable θ, we extend the interpolating G^2 involute curve in the previous section. The radius $\rho(\theta)$ is C^1 continuous. The arclength s is cubic with respect to θ, since $\rho = ds/d\theta$ and $\rho(\theta)$ is quadratic. In this case, we get the indefinite integral

$$\boldsymbol{r} = \int \rho \boldsymbol{t} d\theta = -\rho \boldsymbol{n} + \mu \boldsymbol{t} + \nu \boldsymbol{n}, \tag{9}$$

$$\mu \equiv \frac{d\rho}{d\theta}, \quad \nu \equiv \frac{d^2\rho}{d\theta^2} = const.$$

Using (9) span by span, we derive the continuity conditions

$$\rho_i \boldsymbol{n}_i - \rho_{i+1} \boldsymbol{n}_{i+1} - \mu_i \boldsymbol{t}_i + \mu_{i+1} \boldsymbol{t}_{i+1} - \nu_i \boldsymbol{n}_i + \nu_i \boldsymbol{n}_{i+1} = \Delta \boldsymbol{p}_i, \; i = 0,\ 1, \ldots, n-1, \tag{10}$$

$$\rho_i = \rho(\theta_i), \quad \mu_i = \left.\frac{d\rho}{d\theta}\right|_{\theta=\theta_i}, \quad i = 0,\ 1, \ldots, n,$$

$$\nu_i = \left.\frac{d^2\rho}{d\theta^2}\right|_{\theta=\theta_i}, \quad i = 0,\ 1, \ldots, n-1.$$

Based on these conditions, we are going to derive a set of equations with unknown parameters of $\rho(\theta)$ and solve. We use the following notation in $\rho - \theta$ space:

1) Knots: $\theta_{-1}, \theta_0, \ldots, \theta_{n+1}$, where the end knots are of multiplicity 2.

$$\theta_{-1} = \theta_0, \quad \theta_{n+1} = \theta_n.$$

2) de Boor ordinates [2]: $d_0, d_1, \ldots, d_{n+1}$, where d_i corresponds to the Greville abscissa $(\theta_{i-1} + \theta_i)/2$.
3) Bézier ordinates [2]: $b_0, b_1, \ldots, b_{2n}$.

For easy manipulation, we break down the non-uniform B-spline function $\rho(\theta)$ into the following quadratic Bézier functions with local parameter t:

$$\rho(\theta) = (1-t)^2 b_{2i} + 2(1-t)t b_{2i+1} + t^2 b_{2i+2}, \tag{11}$$

$$0 \le t = \frac{\theta - \theta_i}{\Delta\theta_i} \le 1, \quad i = 0, 1, \ldots, n-1.$$

$$b_{2i} = \frac{d_{i+1}\Delta\theta_{i-1} + d_i \Delta\theta_i}{\Delta\theta_{i-1} + \Delta\theta_i}, \quad i = 0,\ 1, \ldots, n,$$

$$b_{2i+1} = d_{i+1}, \quad i = 0,\ 1, \ldots, n-1.$$

From (11), we obtain

$$\rho_i = b_{2i}, \quad \mu_i = \frac{2\Delta b_{2i}}{\Delta\theta_i}, \quad i = 0,\ 1, \ldots, n,$$

$$\nu_i = \frac{2\Delta^2 b_{2i}}{(\Delta\theta_i)^2}, \quad i = 0,\ 1, \ldots, n-1.$$

We solve the equations (10) with unknowns $\theta_0, \theta_1, \ldots, \theta_n, d_0, d_1, \ldots, d_{n+1}$ by the Newton-Raphson method, using initial values from the conventional C^2 cubic splines. The number of equations is $2n$, while the number of unknowns is $2n + 3$. Accordingly we can give 3 more additional requirements. Since the radius of curvature $\rho(\theta)$ determines a unique curve shape, we can specify and control an interpolating curve by the control polygon (Greville abscissae and de Boor ordinates) of $\rho(\theta)$. Therefore the curve includes circular arc, circle involute and involute of the circle involute because $\rho(\theta)$ is a quadratic B-spline function.

Fig. 3 shows an example of an interpolating G^3 involute curve with the same data points and the same end tangents as in Fig. 2. The computation converged after four iterations. The evolute of this curve has cusp points within segments, since the radius of curvature is quadratic, while the evolute in Fig. 2 has cusp points only at junction points. The evolute of the evolute (circular arcs) has three cusp points. The curvature profile shows the smoothness of this G^3 involute curve.

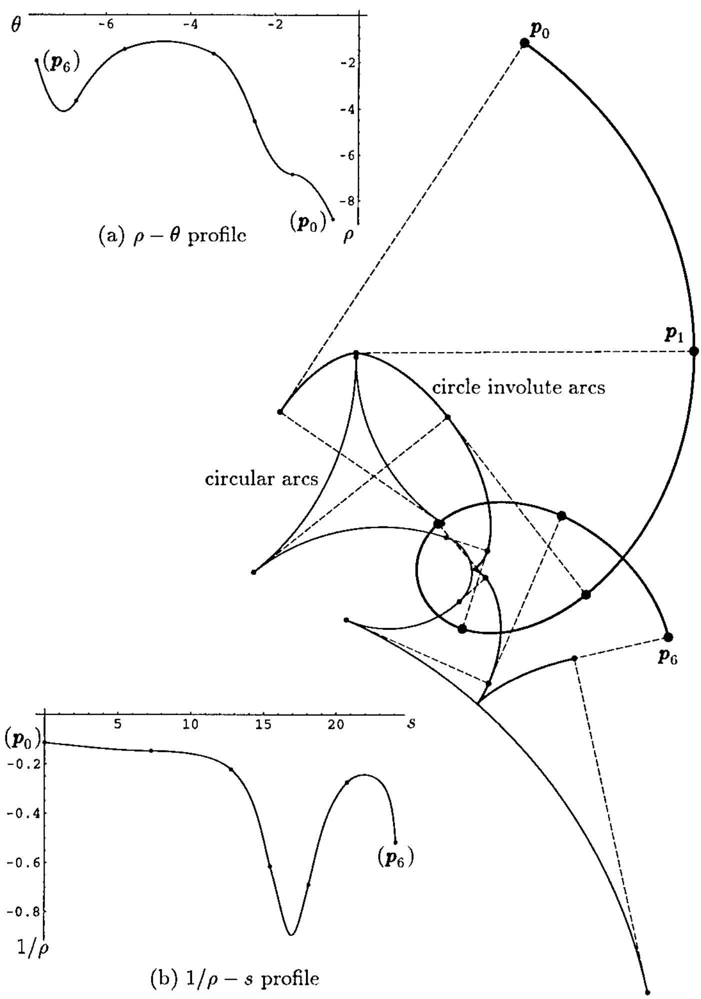
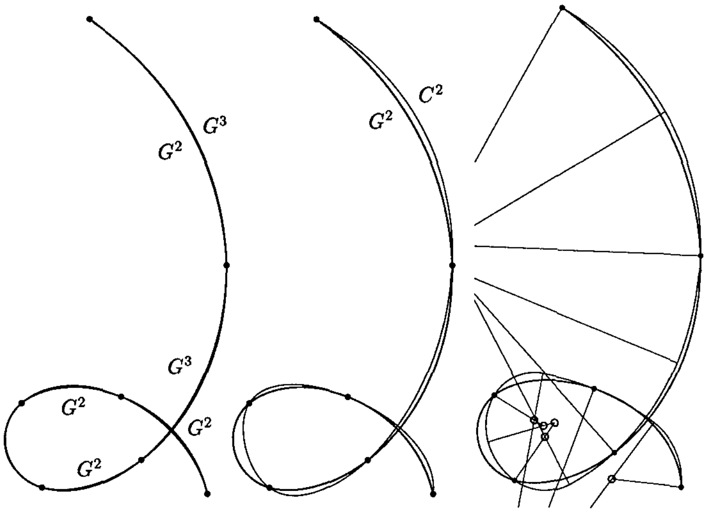


Fig. 3. Interpolating G^3 involute curve.

§5. Some More Numerical Results

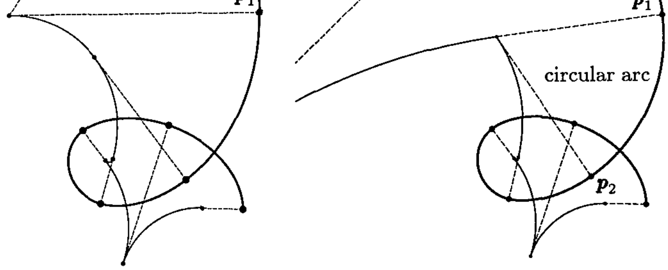
To illustrate the properties of the newly developed curve, we show some more examples of the curves. The same data points are used in Figs. 1 to 5.

The G^2 involute curve is practically more important than the G^1 and G^3 ones. Accordingly, in Fig. 4 the G^2 involute in Fig. 2 is compared with other curves: (a) the G^3 involute curve with the same end tangents in Fig. 3, (b) the conventional C^2 cubic spline curve which is used as an initial curve by the Newton-Raphson method, and (c) a G^1 biarc curve derived by minimum difference between curvatures of two arcs [4]. The labels "G^2","G^3" or "C^2" in Fig. 4 point out which side the corresponding curve passes through. Small circles in Fig. 4(c) are centers of circular arcs. It is understood from



(a) with a G^3 involute (b) with a C^2 cubic curve (c) with a G^1biarc curve

Fig. 4. G^2 involute and comparison with other curves.



(a) a circular arc $\boldsymbol{p}_0\boldsymbol{p}_1$ (b) a circular arc $\boldsymbol{p}_1\boldsymbol{p}_2$

Fig. 5. G^2 involute including a circular arc.

observation that the involute curves are quite smooth.

Fig. 5 illustrates a G^2 involute curve with additional requirements, which includes (a) a circular arc $\boldsymbol{p}_0\boldsymbol{p}_1$ and (b) a circular arc $\boldsymbol{p}_1\boldsymbol{p}_2$.

§6. Concluding Remarks

We proposed the design method of up to G^3 interpolating curve as an involute of circular arcs or an involute of the circle involute arcs. This straightforward approach provides a tool for the construction of planar curves consisting of segments with monotone curvature radius plots of constant sign. Available NC machines with an involute generator are able to draw the objective curves with reduced data.

References

1. FANUC LTD, Involute interpolation (G02.2, G03.2), *FANUC Series 15i-MA Manual B-63324EN/01*, 92–100.
2. Farin, G., *Curves and Surfaces for Computer Aided Geometric Design,* Academic Press, NY, 1988.
3. Farouki, R. T. and J. Rampersad, Cycles upon cycles: an anecdotal history of higher curves in science and engineering, in *Mathematical Methods for Curves and Surfaces II,* Morten Dæhlen, Tom Lyche, Larry L. Schumaker (eds), Vanderbilt University Press, Nashville & London, 1998, 95–116.
4. Su, B.-Q. and D.-Y. Liu, *Computational Geometry: Curve and Surface Modeling,* G.-Z. Chang (Trans.), Academic Press, San Diego, 1989.
5. Toshiba Machine Co., Ltd., Involute interpolation (G105), *TOSNUC 888 Programing Manual (Additional) STE 42864-11*, 68–79.

Mitsuru Kuroda
Toyota Technological Institute
2-12 Hisakata, Tempaku, Nagoya 468-8511, Japan
`kuroda@toyota-ti.ac.jp`

Shinji Mukai
Maebashi Institute of Technology
460-1 Kamisanaru, Maebashi, Gunma 371-0816, Japan
`mukai@maebashi-it.ac.jp`